IMPLEMENTASI *FINE-TUNING* UNTUK PREDIKSI KALIMAT SOLUSI DARI KALIMAT MASALAH PADA ARTIKEL ILMIAH MENGGUNAKAN MODEL *LARGE LANGUAGE MODELS* (LLM)

LAPORAN TUGAS AKHIR

Laporan ini disusun guna memenuhi salah satu syarat untuk menyelesaikan program studi Teknik Informatika S-1 pada Fakultas Teknologi Industri Universitas Islam Sultan Agung

Disusun Oleh:

ALFIYATU NURINAYAH

32602100020

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM SULTAN AGUNG

SEMARANG

2025

IMPLEMENTATION OF FINE-TUNING TO PREDICTE SOLUTION SENTENCES FROM PROBLEM SENTENCES IN SCIENTIFIC ARTICLES USING LARGE LANGUAGE MODELS (LLM) MODEL

FINAL PROJECT

This report was prepared to fulfill one of the requirements for completing the Undergraduate Informatics Engineering study program at the Faculty of Industrial Technology, Sultan Agung Islamic University.

Arranged by:

ALFIYATU NURINAYAH

32602100020

MAJORING IN INFORMATICS ENGINEERING

FACULTY OF INDUSTRIAL TECHNOLOGY

SULTAN AGUNG ISLAMIC UNIVERSITY

2025

LEMBAR PENGESAHAN
TUGAS AKHIR

IMPLEMENTASI *FINE-TUNING* UNTUK PREDIKSI KALIMAT SOLUSI DARI KALIMAT MASALAH PADA ARTIKEL ILMIAH MENGGUNAKAN *LARGE LANGUAGE MODELS* (LLM)

ALFIYATU NURINAYAH
NIM 32602100020

Telah dipertahankan di depan tim penguji ujian sarjana tugas akhir
Program Studi Teknik Informatika
Universitas Islam Sultan Agung
Pada tanggal : 25 Februari 2025

TIM PENGUJI UJIAN SARJANA :

Arief Marwanto, ST, M.Eng, Ph.D
NIDN. 0628097501
(Ketua Penguji) 7/03-2025

Mustafa, ST, MM., M.Kom
NIDN. 0623117703
(Anggota Penguji) 10/03-2025

Sam Farisa C.H., ST, M.Kom
NIDN. 0628028602
(Pembimbing) 10/03-2025

Semarang, 10 Maret 2025

Mengetahui,

Kaprodi Teknik Informatika
Universitas Islam Sultan Agung

Moch. Taufik, S.T.,M.IT
NIDN. 0622037502

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Alfiyatu Nurinayah

NIM : 32602100020

Judul Tugas Akhir : Implementasi *Fine-Tuning* Untuk Prediksi Kalimat Solusi dari Kalimat Masalah pada Artikel Ilmiah Menggunakan *Large Language Models* (LLM)

Dengan ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apbila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 12 Maret 2025

Yang Menyatakan,

Alfiyatu Nurinayah

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama	: Alfiyatu Nurinayah
NIM	: 32602100020
Program Studi	: Teknik Informatika
Fakultas	: Teknologi industri

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : Implementasi *Fine-Tuning* Untuk Prediksi Kalimat Solusi dari Kalimat Masalah Pada Artikel Ilmiah Menggunakan *Large Language Models* (LLM)

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 12 Maret 2025

Yang menyatakan,

Alfiyatu Nurinayah

KATA PENGANTAR

Segala puji dan syukur saya panjatkan ke hadirat Allah SWT atas limpahan rahmat dan karunia-Nya, sehingga saya dapat menyelesaikan Tugas Akhir ini dengan judul "Implementasi *Fine-Tuning* untuk prediksi kalimat solusi dari kalimat masalah pada artikel ilmiah menggunakan LLM". Tugas Akhir ini disusun sebagai salah satu syarat kelulusan dalam menempuh studi serta untuk memperoleh gelar sarjana (S-1) pada Program Studi Teknik Informatika, Fakultas teknologi Industri, Universitas Islam Sultan Agung Semarang.

Dalam penyusunan Tugas Akhir ini, saya mendapatkan banyak bantuan, baik dalam aspek materi maupun teknis dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, saya ingin mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, SH., M.Hum yang mengizinkan penulis menimba ilmu di kampus ini
2. Dekan Fakultas Teknologi Industri Ibu Dr. Ir. Hj. Novi Marlyana, S.T., M.T., IPU., ASEAN.Eng
3. Dosen Pembimbing Bapak Sam Farisa Chaerul Haviana, ST, M.Kom yang telah meluangkan waktu, membimbing, dan memberi ilmu.
4. Orang tua penulis, Bapak Bukhori dan Ibu Sri Nuryati yang selalu memberikan restu, dukungan, serta Doa dalam menyelesaikan Tugas Akhir ini.
5. Para sahabat dan teman-teman atas bantuan, semangat, dan dukungannya.
6. Dan kepada semua pihak yang tidak dapat saya sebutkan satu persatu.

Dengan rendah hati, penulis menyadari bahwa laporan masih memiliki banyak kekurangan dalam kualitas dan isi. karena itu, penulis mengharapkan kritikan dan saran yang membangun untuk penyempurnaan di masa depan.

Semarang, 13 Maret 2025

Alfiyatu Nurinayah

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Prediksi solusi kalimat masalah dalam artikel ilmiah merupakan suatu tantangan karena tidak semua kalimat masalah dalam artikel menyebutkan solusi secara eksplisit. Oleh karena itu, penelitian ini dilakukan untuk mengembangkan model yang mampu memprediksi solusi kalimat masalah dengan menerapkan *fine-tuning* pada model Llama 3.2 1B Instruct. Model ini dilatih menggunakan 99 data latih yang berisi pasangan kalimat masalah dan solusi untuk memahami pola hubungan keduanya. Evaluasi dengan metrik ROUGE menunjukkan bahwa model mencapai skor yang cukup baik, dengan skor ROUGE-1 sebesar 0,611, ROUGE-2 sebesar 0,496, dan ROUGE-L sebesar 0,576. Selama proses pelatihan, nilai *loss* mengalami penurunan yang cukup signifikan yaitu dari 0.4394 pada epoch pertama menjadi 0.0286 pada epoch kelima yang menandakan bahwa model semakin memahami pola pada data pelatihan. Namun penelitian ini menghadapi beberapa kendala, seperti ketidakmampuan model dalam memprediksi solusi beberapa kalimat secara akurat, terutama jika permasalahan yang diberikan tidak berada dalam cakupan data pelatihan. Selain itu, ketika model diterapkan dalam aplikasi, hasil prediksi terkadang kurang relevan, kemungkinan besar disebabkan oleh keterbatasan komputasi pada perangkat yang digunakan. Oleh karena itu, diperlukan penelitian lebih lanjut untuk meningkatkan akurasi model dan mengoptimalkan kinerjanya agar lebih stabil ketika diterapkan pada sistem dengan sumber daya terbatas.

Kata Kunci : *Fine-tuning*, Llama3.2 1B Instruct, Prediksi Solusi, Artikel Ilmiah, NLP.

ABSTRACT

Many articles present a problem without explicitly mentioning the solution, making it difficult to automatically extract relevant answers. This research aims to predict solutions to problem formulations in scientific articles using fine-tuning on the Instruct Llama 3.2 1B model. The model was trained on 99 problem solution sentence pairs, with performance evaluated using the ROUGE metric. The research results show ROUGE-1: 0.611, ROUGE-2: 0.496, and ROUGE-L: 0.576, indicating a moderate ability to produce relevant solutions. During training, the loss decreases significantly from 0.4394 in the first epoch to 0.0286 in the fifth, indicating effective learning. However, the model faces challenges in accurately predicting certain sentences and generating relevant solutions when faced with out-of-scope input problems. Additionally, when integrated into applications, model responses are sometimes inconsistent due to hardware limitations, affecting computational efficiency. Future improvements may include expanding the dataset, refining pre-processing techniques, and optimizing fitting strategies to improve model generalization and accuracy.

Keywords: Fine-tuning, LLaMA 3.2 1B Instructions, Solution Prediction, Scientific Articles, NLP.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Artikel ilmiah merupakan sumber utama pengembangan ilmu pengetahuan, yang berperan penting dalam menyampaikan hasil penelitian, identifikasi masalah, dan solusi yang diusulkan. Namun, di dalam artikel ilmiah tidak semua menyajikan solusi secara eksplisit untuk setiap permasalahan yang dibahas. Banyak artikel ilmiah hanya menggambarkan masalah atau analisis tanpa memberikan alternatif penyelesaian, sehingga pembaca, khususnya akademisi dan peneliti, harus berupaya lebih keras untuk menemukan solusi yang relevan.

Masalah ini menjadi semakin kompleks seiring dengan meningkatnya jumlah artikel ilmiah yang diunggah setiap tahunnya. Berdasarkan data Database Jurnal Ilmiah Indonesia (ISJD), pada tahun 2017 terdapat 23.980 jurnal yang diunggah, dengan jumlah jurnal yang dibaca mencapai 23.910, dan 23.912 jurnal yang diunduh. Pada tahun 2018, jumlah jurnal yang diunggah meningkat signifikan menjadi 34.551 jurnal, dengan rincian 34.231 jurnal dibaca dan 34.215 jurnal diunduh (Fatmalasari dan Lumbanraja 2022). Lonjakan jumlah artikel ini mencerminkan kemajuan dalam produksi ilmu pengetahuan, namun di sisi lain menimbulkan tantangan besar dalam menavigasi informasi yang tersebar luas, terutama untuk mencari solusi atas permasalahan tertentu yang mungkin tidak secara eksplisit disebutkan dalam teks artikel.

Ollama merupakan inovasi di bidang pengolahan bahasa alami yang telah memberikan kontribusi besar dalam memahami dan menganalisis teks. Ollama merupakan *framework* yang digunakan untuk menjalankan model bahasa besar (LLM) secara lokal (Peña dan Ortega-Castro 2024). Model-model yang dikembangkan oleh Ollama, seperti Llama-3.2, adalah contoh dari LLM yang memiliki kemampuan dalam menangani berbagai tugas NLP. Llama-3.2 dirancang dengan arsitektur transformator yang sangat efisien untuk berbagai tugas pemrosesan bahasa alami, termasuk prediksi solusi dari kalimat permasalahan.

Llama-3.2 menggunakan pendekatan *auto-regresif*, di mana *input* teks diproses untuk menghasilkan *output* yang *relevan* dan berkualitas tinggi. Model ini mampu menangani tugas-tugas kompleks seperti penulisan ulang, ringkasan teks, dan bahkan menjawab pertanyaan berdasarkan teks yang diberikan. Meskipun LLM dalam bentuk generik memiliki kemampuan yang luas, sering kali perlu dioptimalkan untuk tugas-tugas tertentu, seperti identifikasi kalimat solusi dalam artikel ilmiah.

Llama-3.2-1B-Instruct adalah varian yang dioptimalkan dari Llama-3.2 untuk tugas-tugas instruksional dan kontekstual melalui proses *fine-tuning*. Pendekatan *fine-tuning* pada LLM menawarkan solusi untuk mengatasi kemampuan yang luas tersebut. Teknik pengoptimalan model pembelajaran mesin yang digunakan untuk meningkatkan performa model, Khususnya dalam konteks memprediksi kalimat solusi dalam artikel ilmiah, proses ini bekerja dengan menyesuaikan *parameter* model dan konstanta penyesuaian dalam model yang telah dilatih sebelumnya agar sesuai dengan tugas baru. (Pradana, Setiadi, dan Muslikh 2024).

Penelitian ini bertujuan untuk mengimplementasikan *fine-tuning* pada model LLM guna memprediksi kalimat solusi dari masalah yang disajikan dalam artikel ilmiah. Dalam pengembangan model prediksi kalimat solusi, *fine-tuning* menjadi sangat relevan karena memungkinkan adaptasi model bahasa besar (LLM) yang telah dilatih dengan dataset yang luas untuk dioptimalkan pada tugas yang lebih spesifik. Dengan mengembangkan penelitian ini, diharapkan dapat membantu pembaca untuk mendapatkan kalimat solusi, sehingga memudahkan peneliti, akademisi, atau pembaca lainnya dalam mengakses solusi dari berbagai masalah yang dibahas dalam artikel ilmiah.

1.2 Perumusan Masalah

Bagaimana penerapan *fine-tuning* pada model berbasis *Large Language Models* (LLM) dapat dioptimalkan untuk memprediksi kalimat solusi dalam artikel secara akurat dan efisien?

1.3 Pembatasan Masalah

Adapun pembatasan masalah pada penelitian ini, yaitu :

1. Model *Large Language Models* (LLM) yang digunakan adalah llama 3.2 1B Instruct
2. Prenelitian ini hanya berfokus pada kalimat dari artikel ilmiah berbahasa Indonesia
3. Dataset yang digunakan hanya diambil dari artikel ilmiah yang sudah dipublikasikan dan di*review* secara manual oleh peneliti dan divalidasi oleh dosen pembimbing
4. Dataset yang digunakan hanya diambil dari artikel ilmiah tentang Teknologi
5. Dataset yang digunakan hanya terdiri dari 4 kategori yaitu, Masalah-Solusi, Tantangan-Jawaban, Peluang-Jawaban, dan Kelemahan-Peringatan.

1.4 Tujuan

Membuat sistem untuk menghasilkan kalimat solusi dari masalah pada artikel ilmiah dengan mengimplementasikan *fine-tuning* menggunakan LLM.

1.5 Manfaat

Manfaat dari Tugas Akhir ini adalah sebagai berikut :

1. Membantu pembaca, peneliti, dan akademisi untuk mengakses solusi dalam artikel ilmiah secara lebih cepat dan efisien .
2. Meningkatkan pemahaman dan akurasi hubungan antara kalimat masalah dan solusi melalui pengguna model llama3.2 1B Instruct yang telah di*fine-tune*

1.6 Sistematika Penulisan

BAB I : PENDAHULUAN

Pada bab ini, penulis mengutarakan urgensi dari penelitian yang dibahas, mulai dari penyusunan latar belakang, menyusun perumusan masalah, menentukan lingkup pembahasan, serta tujuan dan manfaat yang didapat, dan sistematika penulisan

BAB II : TINJAUAN PUSTAKA DAN DASAR TEORI

Bab ini berisi landasan teori yang digunakan, serta referensi dari penelitian sebelumnya yang mendukung perancangan sistem. Selain itu, bagian ini membantu penulis untuk memahami konsep yang berkaitan dengan *Fine-Tuning*, LLM, dan Ollama selama proses penelitian.

BAB III : METODE PENELITIAN

Pada bab 3, penulis menjelaskan tahapan penelitian, mulai dari pengumpulan dataset hingga pemodelan topik berdasarkan data yang tersedia.

BAB IV : HASIL DAN ANALISIS PENELITIAN

Pada bab 4, penulis memaparkan temuan penelitian, yaitu hasil penerapan *fine-tuning* learning untuk prediksi kalimat Solusi dari masalah pada artikel ilmiah menggunakan LLM.

BAB V : KESIMPULAN DAN SARAN

Pada bab 5, penulis menyajikan kesimpulan dari seluruh rangkaian proses penelitian, mulai dari awal hingga akhir.

BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Artikel ilmiah mempunyai peran signifikan dalam mendukung perkembangan ilmu pengetahuan karena dapat menyampaikan hasil penelitian, mengidentifikasi permasalahan dan memberikan solusi. Namun banyak artikel yang tidak secara langsung menyajikan solusi dari permasalahan yang dibahas, sehingga pembaca khususnya peneliti perlu mencari referensi lain untuk menemukan solusi yang sesuai.

Dalam perkembangan teknologi pemrosesan bahasa alami (NLP), LLM terbukti efektif dalam menangani berbagai tugas teks. Dengan pendekatan *text-to-text*, LLM dapat digunakan untuk memprediksi solusi kalimat karena mampu mengelola tugas-tugas NLP yang memerlukan pemahaman dan keluaran teks pada saat yang bersamaan. Namun model tersebut masih perlu diadaptasi melalui penyesuaian untuk tugas yang lebih spesifik, seperti memprediksi solusi dalam kalimat artikel ilmiah. Penyempurnaan memungkinkan model menjadi lebih relevan dengan tugas yang ada.

Adapun penelitian sebelumnya (Mustafid, Pamuji, dan Helmiyah 2020) mengidentifikasi 16 model dengan dua pendekatan yaitu pelatihan konvensional dan pembelajaran transfer yang dilengkapi dengan *fine-tuning*. Hasilnya menunjukkan bahwa *Transfer Learning* memungkinkan pengembangan model *deep learning* dengan akurasi tinggi dalam waktu pelatihan yang lebih singkat, meski dengan jumlah data boneka dan klasifikasi terbatas yang melibatkan banyak kelas. Arsitektur kecil EfficientNetB0 dan MobileNetV3 terbukti lebih unggul, dengan akurasi pelatihan mencapai 100%. Untuk akurasi validasi, EfficientNetB0 mencapai 98,33%, sedangkan MobileNetV3-small sedikit lebih baik dengan 98,75%. Namun *Transfer Learning* juga memiliki kelemahan, yaitu sebagian besar model menunjukkan

variasi data pengujian yang terbatas, sehingga akurasinya lebih rendah. Kondisi ini terkadang menyebabkan model mengalami *overfit* terlalu cepat.

Kemudian penelitian (Alahmari, Atwell, dan Saadany 2024) yang berfokus pada penerjemahan mesin (MT) dari dialek Arab (AD) ke Bahasa Arab Standar Modern (MSA) menggunakan lima kumpulan data bahasa Arab paralel: MADAR, PADIC, Dial2MSA, STS Arab, dan SADID. Dalam penelitian ini, kami mengevaluasi empat varian model AraT5, yaitu AraT5 base, AraT5v2-base-1024, AraT5-MSASmall, dan AraT5-MSA-Base. Hasil eksperimen yang kami sajikan menunjukkan potensi penggunaan metode ini untuk mengotomatisasi konstruksi corpora paralel Arab. Selain itu, penelitian ini juga berkomitmen untuk mendorong kemajuan di bidang tersebut melalui eksplorasi lebih lanjut teknik *fine-tuning* pada model transformator. Ke depan, salah satu arah penelitian yang dapat dikembangkan adalah pengembangan model multibahasa yang dirancang khusus untuk AD dan MSA. Arah lainnya adalah pembentukan korpora Arab paralel tambahan yang mencakup dialek Arab yang kurang terwakili, seperti korpus yang berfokus pada dialek regional Arab Saudi. Selain itu, fenomena Arabizi yaitu penggunaan huruf latin dan angka oleh generasi muda Arab di media sosial untuk mewakili bunyi bahasa Arab juga merupakan aspek penting yang perlu dipertimbangkan dalam penelitian mendatang. Penelitian ini diharapkan dapat menjadi dasar eksplorasi lebih lanjut dalam membangun dan meningkatkan teknologi bahasa Arab, khususnya dalam konteks penerjemahan dan pengolahan dialek.

Kemudian dari hasil penelitian (Zhuang dkk. 2023) yang membahas tentang penerapan model T5 yang telah dilatih sebelumnya untuk tugas pemeringkatan teks. Berbeda dengan monoT5 yang kompatibel dengan T5, penelitian ini memperkenalkan dua varian model berdasarkan arsitektur T5, yaitu model *encoder-decoder* dan model *encoder-only* yang menghasilkan keluaran berupa nilai numerik. Penelitian ini mengusulkan metode penyesuaian T5 menggunakan kerugian peringkas untuk mengoptimalkan metrik peringkas. Hasilnya menunjukkan bahwa penggunaan kerugian peringkat secara signifikan meningkatkan kinerja model T5 ketika disesuaikan dengan dataset MS MARCO

dan NQ. Selain itu, peningkatan kinerja ini juga terbukti konsisten dalam pengujian *zero-shot* pada dataset dari domain berbeda.

Kemudian dari penelitian (Nawrot 2023) memperkenalkan nanoT5, sebuah *framework* PyTorch yang dirancang untuk melakukan pra-pelatihan model T5 secara efisien hanya dengan menggunakan satu GPU. Kerangka kerja ini mampu memberikan kinerja yang sebanding dengan penyiapan skala besar tanpa memerlukan sumber daya yang besar. Penelitian ini juga menyoroti penggunaan pengoptimal AdamW yang dikombinasikan dengan penskalaan RMS untuk meningkatkan stabilitas dan kecepatan pelatihan. Melalui berbagai percobaan, efektivitas framework ini telah terbukti. Tujuan utama penelitian ini adalah menjadikan pelatihan pemodelan bahasa berkualitas tinggi lebih terjangkau sekaligus mendorong kontribusi masyarakat untuk pengembangan lebih lanjut.

Kemudian penelitian (Mengi, Ghorpade, dan Kakade 2023) mengenai penyempurnaan model T5 dan RoBERTa untuk tugas peringkasan teks dan analisis sentimen memberikan banyak wawasan baru serta pencapaian yang cukup signifikan. Model T5 menunjukkan peningkatan yang nyata dalam kemampuannya menghasilkan ringkasan yang lebih koheren dan ringkas. Hal ini terlihat dari peningkatan skor ROUGE-L dari 13.6168 menjadi 21.4997 dan penurunan nilai loss dari 3.0320 menjadi 2.1880 setelah lima *epoch*. Namun, meskipun temuan ini menunjukkan peningkatan performa yang signifikan, optimasi tambahan masih diperlukan untuk mengatasi kemungkinan *overfitting* serta meningkatkan kemampuan model dalam menangani data yang belum pernah ditemui sebelumnya. Di sisi lain, model RoBERTa menunjukkan hasil yang sangat baik dalam tugas analisis sentimen. Dengan akurasi 94,71% dan skor F1 94,60%, model ini terbukti mampu memahami ekspresi sentimen kompleks dalam teks dengan sangat baik. Kombinasi kedua model ini, yang masing-masing dioptimalkan untuk keunggulannya, memberikan pendekatan yang solid untuk mengatasi tantangan besar dalam merangkum dan menganalisis sentimen dalam teks yang panjang. Secara keseluruhan, penelitian ini memberikan kontribusi penting pada bidang NLP, khususnya dalam menunjukkan bagaimana model transformator tingkat lanjut seperti T5 dan RoBERTa dapat diimplementasikan secara efektif untuk

tugas-tugas seperti peringkasan teks dan analisis sentimen. Hasil penelitian ini diharapkan dapat menjadi dasar penelitian selanjutnya di masa yang akan datang, terutama dalam penerapan pada kasus-kasus yang lebih kompleks.

Kemudian dari penelitian (Subarkah, Hilda, dan Zukhronah 2022) menyimpulkan terdapat 10 destinasi wisata dengan jumlah ulasan pengunjung terbanyak yaitu Jln. Malioboro, Tamansari Istana Air, Keraton Yogyakarta, Taman Pintar Yogyakarta, Tugu Yogyakarta, Tebing Breksi, Alun-Alun Selatan, Pantai Parangtritis, Gembira Loka, dan Benteng Vredeburg. Di antara tempat-tempat tersebut, Jalan Malioboro masih menjadi destinasi favorit wisatawan dengan antusiasme tinggi untuk berfoto. Penelitian ini merekomendasikan kepada pemerintah dan pengelola objek wisata di Yogyakarta untuk meningkatkan kualitas fasilitas dan spot foto agar dapat menarik lebih banyak pengunjung. Analisis juga menunjukkan bahwa sebagian besar sentimen negatif pengunjung disebabkan oleh kondisi dan pelayanan yang kurang memadai di tempat wisata. Model *Neural Network IndoBERTweet* diterapkan dengan parameter learning rate sebesar $5x10^{-8}$, jumlah epoch kereta adalah 3, dan langkah penyimpanan adalah 200, menghasilkan akurasi sebesar 92,84%, perolehan rata-rata tertimbang sebesar 93%, presisi sebesar 92%, dan Skor F1 sebesar 92%. Model ini dapat dijadikan acuan oleh pemerintah dan pengembang pariwisata dalam merumuskan kebijakan untuk meningkatkan pengalaman dan kenyamanan pengunjung di Yogyakarta.

Kemudian dari Penelitian (Liu dkk. 2021) memperkenalkan EncT5, sebuah arsitektur berbasis encoder yang dirancang untuk meningkatkan efisiensi saat menyempurnakan pos pemeriksaan T5 tanpa mengorbankan kinerja. Dengan menghilangkan bagian *decoder*, jumlah parameter dalam model berkurang, sehingga latensi inferensi dapat ditingkatkan melalui penggunaan ukuran batch yang lebih besar. Peningkatan throughput ini memberikan manfaat yang signifikan, terutama bagi siswa berprestasi yang ingin melakukan penyulingan dari T5. EncT5 memungkinkan anotasi lebih banyak data tanpa pengawasan, memanfaatkan sumber daya yang sama.

Kemudian dari hasil penelitian (Jayadianti dkk. 2022) menunjukkan bahwa *fine-tuning IndoBERT* yang dikombinasikan dengan RCNN memiliki performa

yang lebih unggul dibandingkan dengan IndoBERT standar dalam analisis sentimen resensi berbahasa Indonesia. Metode ini mencapai hasil terbaik dengan akurasi 95,16% dan skor F1 93,27% menggunakan ukuran batch 16, kecepatan pembelajaran sebesar $3x10^{-5}$. Temuan ini menunjukkan bahwa fine-tuning IndoBERT dengan memanfaatkan seluruh output sebagai masukan untuk RCNN mampu meningkatkan kinerja model dalam analisis sentimen ulasan berbahasa Indonesia. Ke depannya, baik IndoBERT maupun IndoBERT dengan RCNN dapat digunakan untuk berbagai tugas pemrosesan bahasa alami lainnya, seperti deteksi hoax, deteksi sarkasme, klasifikasi berita, analisis topik, klasifikasi emosi, analisis sentimen berbasis aspek, penandaan pos, pengenalan entitas bernama , menjawab pertanyaan, dan tugas NLP lainnya.

Kemudian dari hasil penelitian (Mastropaolo dkk. 2021) mengeksplorasi penggunaan model T5 untuk mendukung empat tugas terkait kode, yaitu perbaikan bug otomatis, pembuatan pernyataan tegas dalam metode pengujian, peringkasan kode, dan injeksi mutan kode. Hasilnya menunjukkan bahwa model T5 mampu menjalankan tugas tersebut dengan sangat baik, bahkan melampaui kinerja empat baseline yang digunakan sebagai pembanding. Namun seperti yang dijelaskan pada bagian V-F, Penelitian lebih lanjut diperlukan untuk mengeksplorasi berbagai faktor mendukung performa superior model T5. Selain itu, (Raffel dkk. 2020) yang pertama kali memperkenalkan model T5 mengungkapkan bahwa versi T5 yang lebih besar dapat memberikan hasil yang jauh lebih baik dibandingkan model T5 kecil yang digunakan dalam penelitian ini. Oleh karena itu, hasil yang dilaporkan pada riset ini dapat dianggap sebagai batas bawah potensi kemampuan T5. Seluruh kode dan data yang digunakan dalam penelitian ini tersedia untuk umum, sehingga dapat digunakan untuk penelitian selanjutnya.

Kemudian dari penelitian (Yazid dan Winarko 2023) yang sukses mengembangkan POS *tagger* berbasis *BERT* dengan performa unggul, memperoleh *recall* sebesar 0,965 selama pelatihan dan 0,961 saat pengujian. Selain itu, model ini mampu memberikan pelabelan yang akurat untuk kata-kata ambigu, dengan Tingkat keberhasilan mencapai 96 dari 100 kasus. Proses penyempurnaan model *BERT*, yang melibatkan penyesuaian parameter dan penambahan data pelatihan,

berkontribusi signifikan terhadap peningkatan kinerja model, seperti dapat dilihat dari evaluasi hasil pada setiap percobaan. Namun, perubahan parameter selama penyesuaian menghasilkan peningkatan kompleksitas komputasi, yang memengaruhi durasi pelatihan dan kebutuhan sumber daya. Penggunaan *BERT* untuk penandaan POS menawarkan fleksibilitas yang lebih besar dibandingkan pendekatan berbasis aturan atau probabilistik. Model ini dapat diterapkan pada kumpulan data yang berbeda dengan menyesuaikan dan memperluas data pelatihan agar selaras dengan konteks yang ditargetkan. Meskipun model yang dikembangkan menunjukkan kinerja yang tinggi, namun hasil evaluasi menunjukkan bahwa model masih mengalami kesulitan dalam mengidentifikasi beberapa kata yang memiliki makna ganda dalam pola tertentu, seperti kata tunggal yang tidak membentuk satuan frasa.

2.2 Dasar Teori

2.2.1 *Natural Language Processing* (NLP)

NLP merupakan kemampuan suatu komputer mengenali dan memproses bahasa alami manusia. baik dalam bentuk lisan maupun tulisan, sebagaimana digunakan dalam komunikasi sehari-hari. NLP secara sederhana bertujuan untuk memungkinkan komputer memahami instruksi yang ditulis dalam bahasa manusia (R, Salim, dan Hasnawi 2022). Untuk konteks penelitian ini, NLP digunakan untuk menganalisis teks pada artikel ilmiah, yang bertujuan untuk mendeteksi kalimat yang memuat solusi dari permasalahan yang dibahas. Tugas ini membutuhkan kemampuan pemrosesan teks yang mendalam untuk memahami kebutuhan antara kalimat masalah dan solusi, yang sering kali bersifat implisit. Dengan menggunakan NLP, penelitian ini berfokus pada pengembangan sistem berbasis model besar untuk mengidentifikasi solusi secara otomatis, sehingga mempermudah proses pemahaman artikel ilmiah oleh akademisi dan peneliti.

2.2.2 *Large Language Models* (LLM)

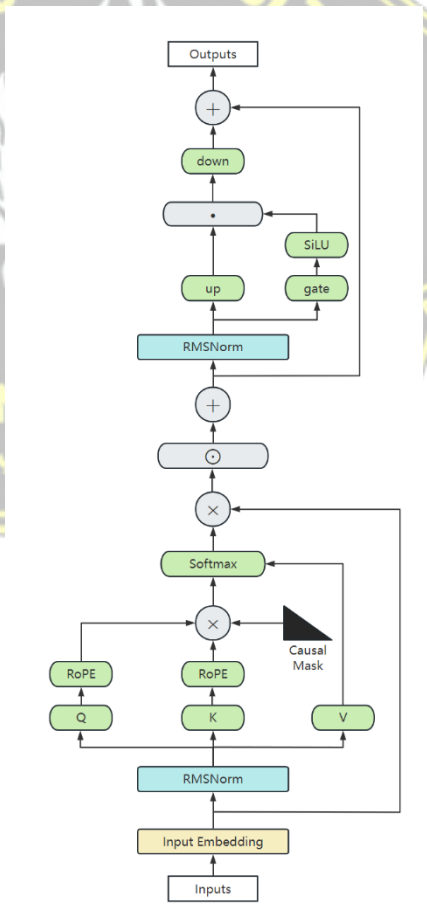
LLM merupakan model bahasa skala besar yang dilatih dengan banyak data teks untuk memahami serta menghasilkan bahasa alami. Model ini diterapkan dalam berbagai bidang, seperti penerjemahan, pembuatan konten, dan *chatbot* untuk dukungan emosional (J dkk. 2024). LLM seperti llama3.2 1B Instruct digunakan untuk menangkap pola hubungan dalam teks artikel ilmiah, termasuk pola hubungan antara kalimat masalah dan solusi. LLM menawarkan keunggulan berupa fleksibilitas untuk menyelesaikan berbagai tugas NLP hanya dengan memodifikasi format *input* dan *output.* Contoh implementasi LLM dalam penelitian ini adalah menghasilkan kalimat solusi dari kalimat masalah melalui pendekatan *text-to-text.* Model ini dilatih secara khusus untuk memahami konteks dalam artikel ilmiah dan menghasilkan kalimat solusi yang relevan, dengan focus pada kategori masalah-solusi, tantangan-jawaban, peluang-jawaban, dan kelemahan-peningkatan.

2.2.3 *Fine – Tuning*

Fine-Tuning merupakan teknik pembelajaran mesin yang umum digunakan untuk menyesuaikan model bahasa yang telah dilatih sebelumnya agar sesuai dengan tugas-tugas lanjutan tertentu. Strategi ini memperbarui bobot model dengan melatihnya menggunakan dataset berlabel yang relevan dengan tugas yang ditargetkan. (Mosbach dkk. 2023). Proses *fine-tuning* memungkinkan model yang awalnya bersifat umum untuk mengkhususkan diri pada tugas tertentu, seperti prediksi kalimat solusi. *Fine-Tuning* diterapkan pada model llama 3.2 1B Instruct yang telah dilatih sebelumnya menggunakan dataset *generic*. Proses *fine-tuning* melibatkan pelatihan ulangg model dengan dataset berupa pasangan kalimat masalah-solusi yang diambil dari artikel ilmiah. Tahapan ini bertujuan agar model dapat mengenali pola hubungan antara masalah dan solusi dalam teks. Dengan *fine-tuning,* model llama 3.2 yang awalnya bersifat umum dioptimalkan untuk tugas prediksi kalimat solusi, menghasilkan prediksi yang lebih relevan dengan konteks artikel ilmiah.

2.2.4 Ollama

Ollama merupakan kerangka kerja yang dirancang untuk menjalankan model bahasa besar langsung di perangkat pengguna. Dengan menggunakan Ollama, model kecerdasan buatan dapat dijalankan secara lokal tanpa memerlukan ketergantungan pada layanan *cloud*, sehingga memberikan fleksibilitas dan efisiensi dalam penggunaannya (Peña dan Ortega-Castro 2024). Ollama mempunyai banyak model di dalam nya, salah satu nya yaitu llama 3.2 1B Instruct yang merupakan koleksi model *generative* yang telah dilatih sebelumnya dan disesuaikan dengan instruksi dalam ukuran 1B dan 3B. Llama juga merupakan model bahasa auto-regresif yang menggunakan arsitektur transformator yang dioptimalkan.



Gambar 2. 1 Arsitektur Llama (Chen dkk. 2024)

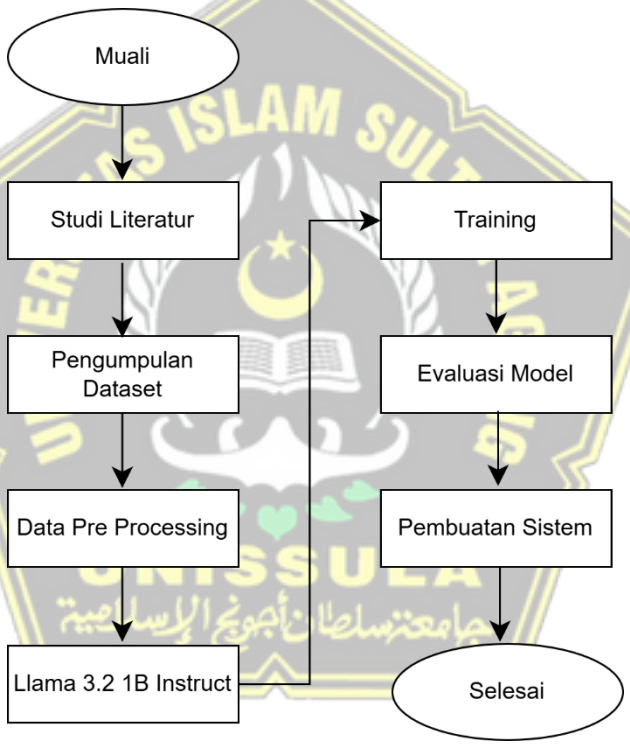
Model Llama dilengkapi mekanisme *Group Query Attention* sebagai strategi untuk mengoptimalkan penggunaan memori, sehingga secara drastis mengurangi kebutuhan komputasi dan memori sekaligus meningkatkan efisiensi operasional. Setelah tahap self-attention, hasilnya akan digabungkan kembali dengan input melalui koneksi sisa, kemudian dilewatkan melalui *feedforward neural network* (FFN) yang terdiri dari dua lapisan linier dan fungsi aktivasi seperti SiLU. Proses ini berperan dalam meningkatkan deteksi fitur *non-linear* sebelum akhirnya melalui tahap normalisasi Final RMSNorm, yang menghasilkan array probabilitas logit untuk memprediksi token berdasarkan strategi seleksi tertentu (Chen dkk. 2024).

Dari perspektif arsitektur, model Llama dirancang dengan pendekatan terstruktur yang efisien dan optimal, memprioritaskan tugas generatif melalui model khusus *Decoder*. Pendekatan ini menyederhanakan proses pelatihan dan meningkatkan kecepatan eksekusi. Dengan penerapan RMSNorm dan teknik pengkodean posisi yang inovatif, stabilitas model meningkat, sekaligus memperkaya kemampuannya untuk mengekspresikan informasi. Hal ini memungkinkan generalisasi yang lebih baik serta efisiensi tinggi dalam menangani kumpulan data besar. Kombinasi mekanisme perhatian yang dirancang secara tepat dan jaringan umpan maju memungkinkan model ini menangkap dan memahami pola bahasa yang kompleks, sehingga sangat andal dalam berbagai tugas pemrosesan bahasa alami (Chen dkk. 2024).

BAB III
METODE PENELITIAN

3.1 Metode Penelitian

Penelitian ini menggunakan metode *Fine-Tuning* menggunakan LLM. Dalam penelitian ini, LLM akan di *training* untuk memprediksi solusi dari kalimat masalah yang terdapat dalam artikel ilmiah. Tahapan yang perlu dilakukan dalam penelitian ini adalah sebagai berikut :



Gambar 3. 1 Alur penelitian

3.1.1 *Studi Literatur*

Peneliti akan meninjau berbagai sumber literatur, termasuk *e-book*, artikel, jurnal, penelitian sebelumnya seperti tesis dan skripsi, serta informasi dari berbagai situs web yang relevan dengan topik penelitian. Studi literatur ini berfokus pada pemahaman teori mengenai *fine-tuning,* LLM, metode prediksi berbasis teks, serta penerapan model *machine learning* untuk menganalisis solusi, tantangan, peluang, dan kelemahan yang diidentifikasi dalam artikel ilmiah.

3.1.2 Pengumpulan Dataset

Proses pengumpulan dataset pada penelitian ini dilakukan secara manual karena belum tersedia dataset yang sesuai dengan kebutuhan penelitian. Dataset ini disusun dengan mereview 100 *artikel*. Artikel yang digunakan diambil dari Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) https://drive.google.com/drive/folders/1XsB0msl-S9PWNlgRFd-tzN0FX1YPn6dF?usp=drive_link .

Tahapan pengumpulan data dilakukan sebagai berikut :

a. Identifikasi Permasalahan dalam Artikel

Proses pengumpulan dimulai dengan mengkaji permasalahan yang terdapat di setiap artikel. Setiap kalimat yang memuat permasalahan dan solusinya dicatat untuk membentuk dataset.

b. Kategori Data

Setiap kategori berisi minimal 20 kalimat permasalahan, dan data yang terkumpul sebanyak 99 kalimat. Dataset dikelompokkan ke dalam empat kategori utama, yaitu :

a) Masalah – Solusi
b) Tantangan – Jawaban
c) Peluang – Jawaban
d) Kelemahan – Peningkatan

c. Pengambilan Kalimat Permasalahan dan Solusi

a) Jika dalam artikel terdapat lebih dari satu kalimat yang menyatakan permasalahan, semua kalimat tersebut dicatat dan dimasukkan ke dalan dataset.
b) Jika artikel memuat kalimat permasalahan beserta solusi nya, keduanya dicatat dan dimasukkan ke dalam dataset.
c) Apabila ditemukan kalimat permasalahan tanpa solusi dalam artikel, solusi dibuat berdasarkan asumsi yang relevan dan diarahkan untuk menyelesaikan masalah tersebut. Asumsi ini kemudian divalidasi oleh dosen pembimbing.

d. Validasi Dataset

Setelah data terkumpul, seluruh dataset diverifikasi oleh dosen pembimbing. Proses validasi bertujuan memastikan bahwa dataset sesuai dengan kebutuhan penelitian dan memiliki kualitas yang memadai.

Sebagai pelengkap, berikut ini disertakan contoh kalimat permasalahan berdasrkan masing-masing kategori.

Tabel 3. 1 Contoh kalimat *problem-solution*

No	Kategori	*Problem*	*Solution*
1	*Problem-Solution*	Pemesanan barang di Apotek Afdhal masih mengandalkan pembukuan manual, yang sering mengakibatkan kesalahan pendataan barang dan keterlambatan pemesanan	Diusulkan penerapan sistem informasi terintegrasi untuk mengotomatisasi proses pemesanan, sehingga dapat meningkatkan efisiensi dan mengurangi kesalahan dalam pendataan barang
2	Tantangan-Jawaban	Transisi ke rekam medis elektronik menjadi langkah penting, namun juga menghadirkan tantangan besar dalam hal pengelolaan data	Dengan menerapkan metode Agile, rumah sakit memiliki kesempatan untuk mengatasi masalah operasional, meningkatkan ketepatan data, dan menyederhanakan proses pelayanan pasien
3	Peluang-Jawaban	*E-commerce* yang berkembang pesat membuka peluang bagi fintech untuk memudahkan pembayaran digital di masa mendatang	Dengan meningkatnya penggunaan layanan pembayaran digital, penyedia fintech dapat memanfaatkan tren ini untuk menarik lebih banyak pengguna dengan menyediakan layanan yang aman dan efisien

4	Kelemahan-Peningkatan	Proses pengelolaan data sensus harian rawat inap yang masih manual rentan terhadap ketidakakuratan data	Apabila metode manual terus digunakan, risiko kehilangan data dan ketidakonsistenan informasi akan semakin besar, sehingga penting untuk beralih ke sistem otomatis yang lebih aman dan efisien

3.1.3 Data *Pre-Processing*

Dalam tahap ini, Peneliti akan melakukan Pre Processing Data, yaitu :

1. *Case folding, case folding* atau normalisasi huruf merupakan salah satu tahapan dalam *text preprocessing* yang bertujuan untuk menyamakan format huruf dengan mengubah seluruh teks menjadi huruf kecil atau huruf kapital. Proses ini dilakukan agar teks lebih mudah dibandingkan dan dianalisis tanpa terpengaruh oleh perbedaan penggunaan huruf besar. *Case folding* yang dilakukan yaitu *text.lower* untuk menyesuaikan teks menjadi huruf kecil semua. Mengubah semua karakter menjadi huruf kecil membantu meningkatkan konsistensi data, membuat analisis dan pemrosesan lebih lanjut menjadi lebih mudah (Salam dkk. 2023).
2. Noise Removal merupakan salah satu langkah dalam pemrosesan awal teks yang bertujuan untuk merapikan teks dengan menghilangkan spasi berlebih. Dengan cara ini, teks menjadi lebih mudah diproses tanpa terpengaruh oleh jumlah spasi yang tidak seragam. Proses ini dilakukan dengan menggunakan ekspresi reguler `re.sub(r'\s+', ' ', text)`, yang berfungsi untuk mengganti spasi berlebih dengan satu spasi saja. Menghapus spasi yang tidak perlu dapat meningkatkan keterbacaan teks dan mempermudah analisis dan pemrosesan lebih lanjut.
3. Whitespace Normalization, Proses ini bertujuan untuk menghapus spasi berlebih dalam teks dengan mengganti semua jenis spasi dengan satu spasi. Menggunakan fungsi re.sub(r'\s+', ' ', text) memungkinkan spasi ekstra dihapus, sehingga menghasilkan teks yang lebih bersih dan lebih mudah dianalisis.

3.1.4 Pemilihan dan Konfigurasi Model

Setelah data siap, langkah selanjutnya adalah memilih dan mengatur model Llama 3.2 1B Instruct untuk proses *fine-tuning.* Model ini dipilih karena kemampuannya dalam memahami serta menghasilkan teks sesuai dengan instruksi yang diberikan. Dalam tahap konfigurasi, beberapa *hyperparameter* seperti *batch size*, *learning rate*, dan jumlah *epoch* ditetapkan agar proses pelatihan berjalan dengan optimal serta menghindari *overfitting*. Selain itu, format *input output* juga disesuaikan, dengan merancang *prompt* tertentu agar model dapat lebih memahami tugasnya dalam memprediksi solusi dari kalimat masalah.

Prompt yang digunakan dalam penelitian ini dirancang untuk memberikan konteks yang jelas kepada model, dengan format seperti "Question: {input_question}\nAnswer: The output not be more than 2 senteces"

Dengan format ini, model dapat memahami bahwa input berupa sebuah pertanyaan atau masalah yang membutuhkan solusi sebagai *output* nya. Selain itu, *output* juga tidak lebih dari 2 kalimat.

3.1.5 *Fine-Tuning* Model

Fine-tuning merupakan proses penyesuaian model yang telah melalui tahapan sebelumnya (*pre-trained* model) agar lebih efektif dalam menyelesaikan tugas tertentu atau menangani data tertentu. Model pra-terlatih umumnya dilatih dengan kumpulan data yang luas dan bervariasi, sehingga mampu memahami berbagai jenis data secara umum (Susilo, Christanti, dan Lauro 2024).

Pada tahap pelatihan, model llama 3.2 1B instruct dilatih menggunakan dataset yang telah diproses sebelumnya. 99 data akan di*training* untuk memastikan model mampu mengenali pola hubungan antara kalimat masalah dan solusi dengan baik.

Fine-tuning model LLama 3.2 1B Instruct pada penelitian ini dilakukan untuk menyesuaikan model agar lebih optimal dalam memprediksi solusi berdasarkan kalimat permasalahan pada artikel ilmiah. Proses ini menggunakan pendekatan pembelajaran *sequence to sequence*, dimana model dilatih untuk memahami hubungan antara input berupa kalimat masalah dan *output* berupa solusi yang sesuai dengan konteks.

Pelatihan dilakukan dengan strategi evaluasi secara berkala untuk memastikan model dapat beradaptasi dengan baik terhadap data yang digunakan. Jumlah *epoch* ditentukan agar model memiliki waktu yang cukup untuk mempelajari pola hubungan antara masalah dan solusi, sedangkan ukuran batch disesuaikan agar pelatihan berjalan lebih efisien. Selain itu, penyesuaian kecepatan pembelajaran dilakukan untuk menjaga keseimbangan antara stabilitas pelatihan dan kecepatan konvergensi model.

3.1.6 Evaluasi Model

Setelah model selesai dilatih, dilakukan validasi menggunakan metrik evaluasi seperti ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*). Metrik ini mengukur kesamaan antara solusi yang diprediksi oleh model dengan solusi yang ada dalam dataset. Validasi bertujuan untuk mengukur relevansi, akurasi, dan kualitas prediksi model.

ROUGE merupakan pendekatan penilaian yang banyak digunakan dalam NLP, khusunya untuk menilai kualitas ringkasan otomatis dan terjemahan mesin. Evaluasi ini dilakukan dengan mengevaluasi perbandingan hasil prediksi solusi yang diperoleh dari sistem dan dibandingkan dengan data acuan, yang disebut *ground truth summarization.* ROUGE bekerja menggunakan pendekatan *intrinsic*, yakni menganalisis tingkat kesamaan antara hasil prediksi dan referensi yang diharapkan.

ROUGE menyediakan beberapa jenis metrik yang digunakan sebagai standar untuk menilai keakuratan prediksi suatu sistem. Berikut ini adalah metrik utama yang sering digunakan:

1. ROUGE-1 (unigram): Mengukur tingkat kemiripan satu kata antara prediksi dan referensi. Metrik ini membantu menilai apakah kata-kata penting dalam solusi yang dihasilkan sistem cocok dengan solusi referensi.

$$ROUGE\text{-}1\ precision = \frac{jumlah\ unigram\ kata\ sama}{jumlah\ total\ unigram\ dalam\ hasil} \quad (1)$$

$$ROUGE\text{-}1\ recall = \frac{jumlah\ unigram\ kata\ sama}{jumlah\ total\ unigram\ dalam\ referensi} \quad (2)$$

$$ROUGE\text{-}1\ f1\text{-}score = 2x\frac{precision\ (ROUGE-1)\ x\ recall\ (ROUGE-1)}{precision\ (ROUGE-1) + recall\ (ROUGE-1)} \quad (3)$$

2. ROUGE-2: Berfokus pada pasangan kata (bigram) untuk mengevaluasi kesamaan konteks antar kata. ROUGE-2 berguna untuk menilai sejauh mana pasangan kata dalam prediksi mampu mencerminkan konteks atau makna dalam referensi.

$$ROUGE\text{-}2\ precision = \frac{jumlah\ bigram\ kata\ sama}{jumlah\ total\ unigram\ dalam\ hasil} \quad (4)$$

$$ROUGE\text{-}2\ recall = \frac{jumlah\ bigram\ kata\ sama}{jumlah\ total\ unigram\ dalam\ referensi} \quad (5)$$

$$ROUGE\text{-}2\ f1\text{-}score = 2x\frac{precision\ (ROUGE-2)\ x\ recall\ (ROUGE-2)}{precision\ (ROUGE-2) + recall\ (ROUGE-2)} \quad (6)$$

3. ROUGE-L (*Longest Common Subsequence (LCS)*): Metrik ini menilai kesamaan berdasarkan urutan kata atau struktur kalimat. ROUGE-L digunakan untuk mengukur seberapa cocok struktur atau pola kalimat dalam solusi prediksi dengan solusi referensi, sehingga dapat menilai kealamian dan strukturnya.

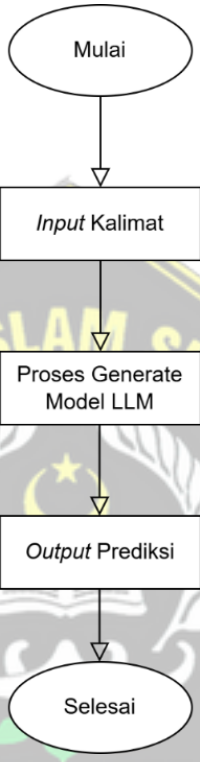
$$ROUGE\text{-}L\ precision = \frac{Longest\ Common\ Subsequence\ (LCS)}{jumlah\ kata\ dalam\ hasil} \quad (7)$$

$$ROUGE\text{-}L\ recall = \frac{Longest\ Common\ Subsequence\ (LCS)}{jumlah\ kata\ dalam\ referensi} \quad (8)$$

$$ROUGE\text{-}L\ f1\text{-}score = 2x\frac{precision\ (ROUGE-L)\ x\ recall\ (ROUGE-L)}{precision\ (ROUGE-L) + recall\ (ROUGE-L)} \quad (9)$$

3.2 Analisis Sistem

Dalam penelitian ini, penulis akan mengembangkan sistem prediksi solusi berbasis *web* yang dirancang untuk menawarkan sistem prediksi solusi bagi akademisi, peneliti, dan banyak orang. Alur sistem akan digambarkan menggunakan flowchart pada Gambar 3.2.



Gambar 3. 2 Alur kerja sistem

Berikut adalah tahapan alur kerja sistem :

a. Mulai

Pengguna terlebih dahulu mengakses halaman utama yang menampilkan proses kerja dari aplikasi sistem prediksi solusi.

b. *Input* Kalimat

Kemudian pengguna memasukkan kalimat permasalahan yang terdapat dalam artikel, setelah itu tekan tombol “Kirim” yang terdapat di tengah *input* teks.

c. Permasalahan diproses

Selanjutnya sistem akan mulai memproses permasalahan yang diberikan oleh user dengan mencocokkan data jawaban yang sudah dilatih.

d. *Output* Prediksi

Setelah permasalahan diproses, model akan memberikan *output* prediksi, yaitu kalimat solusi yang relevan dengan masalah yang diberikan. Solusi ini dihasilkan berdasarkan pola-pola yang telah dipelajari oleh model selama proses *fine-tuning*.

3.3 Analsis Kebutuhan

Pada tahap analisis kebutuhan, peneliti mengevaluasi seluruh perangkat lunak yang dibutuhkan agar sistem prediksi solusi dapat berfungsi dengan optimal dan menghasilkan *output* yang diharapkan. Pengembangan sistem ini menggunakan perangkat lunak berikut :

1. *Python*

   Python merupakan bahasa pemrograman tingkat tinggi yang mudah dibaca dan digunakan untuk berbagai aplikasi, termasuk pemrosesan data dan kecerdasan buatan. Penelitian ini menggunakan *Python* versi 3.12 karena memiliki peningkatan kinerja, efisiensi memori yang lebih baik, dan kompatibilitas dengan perpustakaan terbaru yang diperlukan untuk NLP. Untuk menginstalnya, pengguna dapat mengunduhnya melalui situs resmi https://www.python.org/downloads / , pilih versi 3.12 sesuai sistem operasi, kemudian jalankan file instalasi dengan mencentang opsi "*Add Python to PATH*" agar dapat diakses melalui terminal. Setelah instalasi, dapat diverifikasi dengan menjalankan perintah python --version. Dalam penelitian ini, *Python* digunakan untuk menjalankan berbagai skrip yang berkaitan dengan pemrosesan data, pelatihan model, dan pengembangan aplikasi berbasis *web*.

2. *Google Colaboratory* (Colab)

   Colab adalah layanan berbasis *cloud* dari *Google* yang memungkinkan pengguna menulis dan menjalankan kode *Python* secara interaktif tanpa memerlukan instalasi lokal. *Platform* ini sangat cocok untuk eksperimen pembelajaran mesin karena mendukung penggunaan GPU secara gratis. *Colab* tidak memerlukan instalasi, cukup diakses melalui *browser* di https://colab.research.google.com/ . Dalam penelitian ini, *Colab* digunakan

untuk menulis dan menjalankan kode pemrosesan teks serta melatih model NLP dengan memanfaatkan GPU untuk mempercepat proses *fine-tuning*. Library Transformers

3. *Library Transformers*

   Library Transformers dari *Hugging Face* menyediakan akses ke berbagai model berbasis transformer untuk tugas NLP seperti teks generasi, klasifikasi, dan ekstraksi informasi. Dalam penelitian ini, *Transformers* digunakan untuk memuat model pra-latih dan menyesuaikan model untuk tugas prediksi solusi dari kalimat permasalahan. Untuk menginstalnya, jalankan perintah berikut di terminal atau *command prompt*: pip install transformers. Cara kerjanya adalah dengan menyediakan antarmuka sederhana untuk memuat model besar seperti T5 dan LLaMA, memungkinkan penelitian ini untuk memanfaatkan teknologi NLP terkini dalam memproses teks ilmiah.

4. *Library PyTorch*

   PyTorch digunakan sebagai kerangka kerja utama dalam proses pelatihan dan *inferensi* model berbasis *Natural Language Processing.* Dengan menggunakan *PyTorch*, model dapat dilatih dengan efesiensi menggunakan unit pemrosesan grafis (*GPU*), sehingga mempercepat proses *fine-tuning* model llama3.2 1B Instruct. Instalasi *PyTorch* dapat dilakukan dengan perintah berikut: pip install torch. Namun untuk mendapatkan dukungan akselerasi GPU, instalasi perlu disesuaikan berdasarkan versi CUDA dengan mengikuti petunjuk di https://pytorch.org/get-started/locally/ . Setelah terinstal, *PyTorch* dapat digunakan dengan mengimpornya ke kode *Python* menggunakan: import torch.

5. *Library Pandas*

   Library Pandas adalah perpustakaan *Python* yang digunakan untuk manipulasi dan analisis data dalam format tabel. Dalam penelitiannya, *Pandas* berperan dalam membaca dataset yang berisi pasangan kalimat masalah dan solusi dari artikel ilmiah. Instalasi *Pandas* dapat dilakukan dengan perintah pip install pandas. Setelah terinstal, Pandas dapat digunakan dengan mengimpornya dalam kode *Python* import pandas as pd. Cara kerja Pandas

melibatkan pembacaan dataset dalam format CSV atau lainnya, pembersihan data yang tidak relevan, serta transformasi data agar dapat digunakan dalam proses fine-tuning model NL

6. *Library Numpy* (*Numerical Python*)

   Numpy adalah perpustakaan yang digunakan untuk bekerja dengan *array* serta operasi aljabar linier, transformasi *Fourier*, dan manipulasi matriks. Dalam penelitian ini *NumPy* mendukung pengolahan data dalam bentuk vektor dan matriks yang digunakan dalam pelatihan model NLP. Instalasi *NumPy* dapat dilakukan dengan perintah pip install numpy. Setelah terinstal, perpustakaan ini dapat digunakan dengan mengimpornya ke kode *Python* menggunakan import numpy as np. *NumPy* bekerja dengan menyediakan fungsi pengoptimalan dalam komputasi numerik, mempercepat penghitungan yang diperlukan dalam proses pembelajaran mesin.

7. *Tqdm*

   Library tqdm digunakan untuk menampilkan progress dari proses pelatihan model atau pemrosesan dataset. Dengan *tqdm,* proses yang sedang berjalan dapat dipantau dengan lebih mudah melalui indicator progress yang ditampilkan di terminal. Instalasi tqdm dapat dilakukan dengan menjalankan perintah pip install tqdm. Setelah terinstal, tqdm dapat digunakan dengan mengimpornya ke dalam kode Python menggunakan from tqdm import tqdm. Dalam penelitian ini, tqdm digunakan untuk melacak proses pelatihan model NLP, membantu dalam memantau waktu eksekusi dan kinerja pemrosesan.

8. *ROUGE*

   Library ROUGE digunakan untuk menilai hasil prediksi model dengan membandingkan *output* model dengan solusi acuan. Dalam penelitian ini, metrik *ROUGE* digunakan untuk mengukur sejauh mana solusi prediksi model cocok dengan solusi sebenarnya. Instalasi ROUGE dapat dilakukan dengan perintah pip install rouge-score. Setelah terinstal, *library* ini dapat digunakan dengan mengimpornya dalam kode *Python* menggunakan from rouge_score import rouge_scorer. *Library* ini bekerja dengan menghitung kesamaan n-

gram antara teks prediksi dan referensi, memberikan skor yang mencerminkan kualitas hasil prediksi mode

9. *Visual Studio Code* (VS Code)

   VS Code merupakan editor teks yang digunakan dalam pengembangan aplikasi karena mendukung banyak bahasa dan *framework* pemrograman, serta memiliki ekstensi yang memudahkan pengembangan. Dalam penelitian ini, VS *Code* digunakan untuk menulis dan mengelola kode *Python* yang digunakan dalam pelatihan model dan pengembangan aplikasi *web*. Instalasi dapat dilakukan dengan mendownloadnya dari situs resmi https://code.visualstudio.com/ . Setelah terinstal, pengguna dapat menambahkan ekstensi *Python* untuk mendukung *debugging* serta eksekusi kode langsung di VS Code.

10. *Framework Flask*

    Flask merupakan *framework* berbasis *Python* yang ringan dan *open-source*, dirancang untuk membangun aplikasi berbasis *web* secara fleksibel. Dalam penelitian ini, *framework Flask* digunakan untuk mengembangkan aplikasi *website* untuk menampilkan hasil prediksi solusi dari kalimat masalah yang diolah menggunakan model *fine-tuned*. *Flask* dipilih karena kemudahan integrasinya dengan model pembelajaran mesin dan kemampuannya menangani permintaan API untuk pemrosesan data waktu nyata. Instalasi *Flask* dapat dilakukan dengan menjalankan perintah pip install flask. Setelah terinstal, *Flask* dapat digunakan dengan mengimpornya dalam kode *Python* menggunakan from flask import Flask. Cara kerja *Flask* pada penelitian ini adalah dengan membuat API yang menerima input teks dari pengguna, mengirimkannya ke model NLP, kemudian mengembalikan hasil prediksi dalam format yang dapat diakses melalui *web browser*.

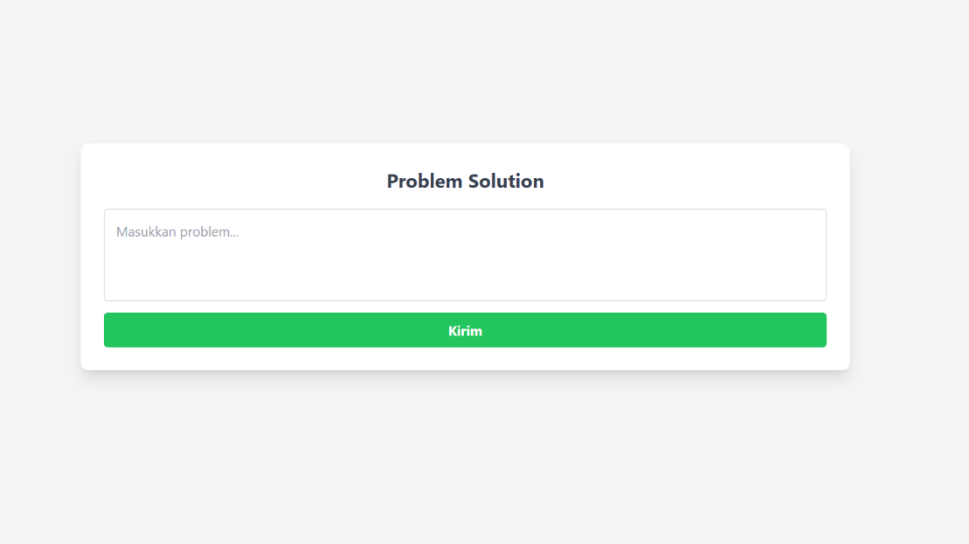
BAB IV
HASIL DAN ANALISIS PENELITIAN

4.1 Hasil Penelitian

Setelah model berhasil diunduh, langkah selanjutnya adalah membuat *User Interface* (UI) agar lebih mudah digunakan untuk mendapatkan prediksi solusi dari masalah yang dimasukkan.

Aplikasi *web* ini dirancang agar pengguna cukup memasukkan kalimat permasalahan pada kolom masukan, kemudian sistem akan memprosesnya dan menampilkan hasil prediksi solusi pada halaman keluaran. Dengan memanfaatkan *Flask*, aplikasi ini dapat dijalankan secara lokal atau diunggah ke *server* untuk memudahkan akses pengguna lain.

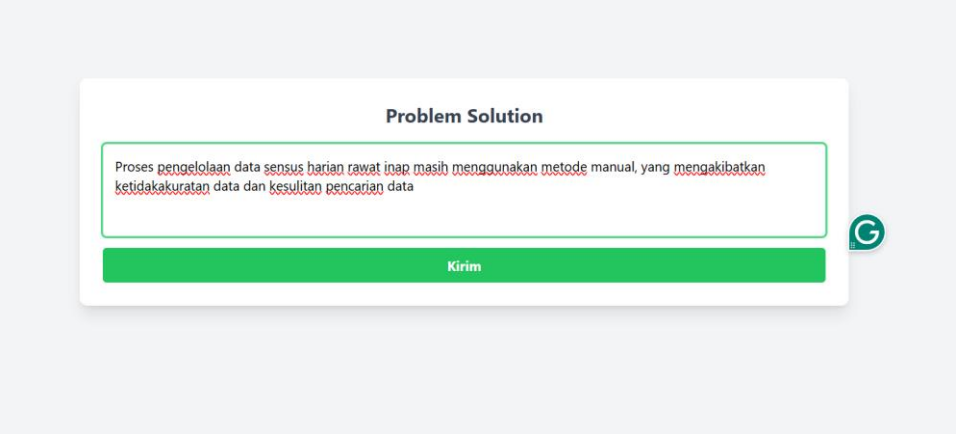
4.1.1 Tampilan Awal Aplikasi



Gambar 4. 1 Tampilan awal UI

Gambar 4.1 menampilkan awal dari demo aplikasi, yang hasil dari penggabungan model dengan skrip pada aplikasi untuk memudahkan pengguna agar dapat menemukan prediksi solusi dari kalimat permasalahn dalam artikel pada sistem *problem-solution.* Dengan menambahkan kalimat permasalahan di kolom masukan *problem,* pengguna akan mendapatkan solusi dengan cara klik tombol “kirim”.

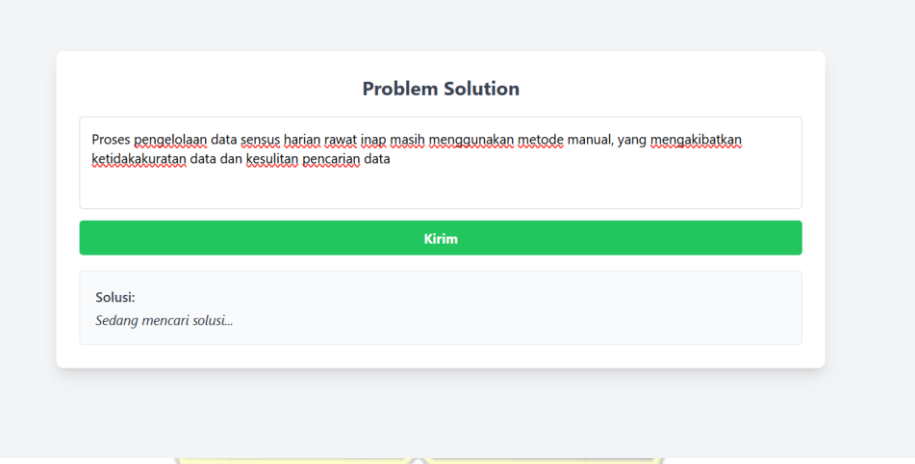
4.2.1 Tampilan *Input Problem*



Gambar 4. 2 Tampilan saat *Input Problem*

Gambar 4.2 menampilkan saat pengguna memasukkan kalimat masalah untuk mendapatkan prediksi.

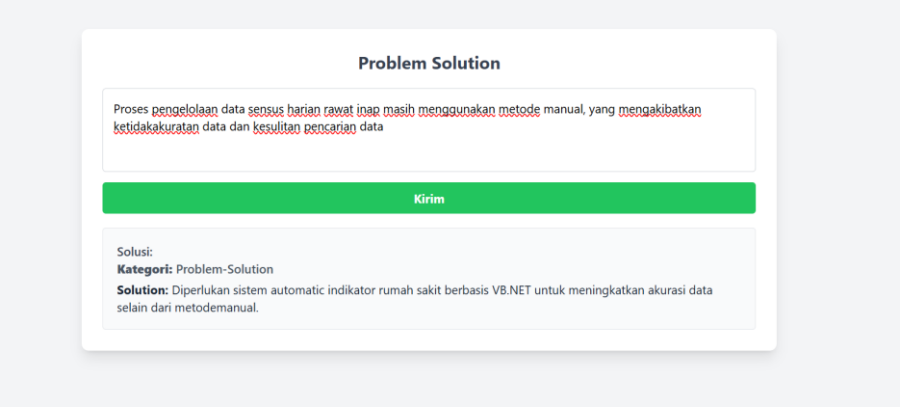
4.3.1 Tampilan Saat sistem *Generate* Model



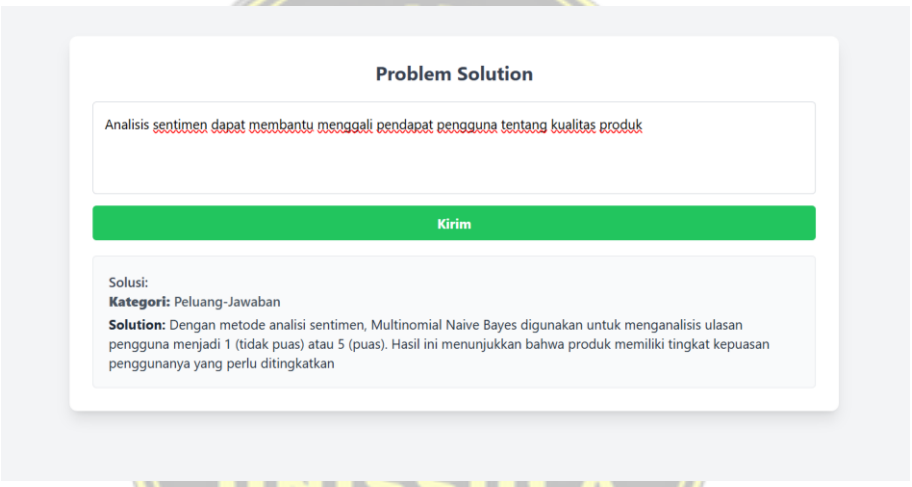
Gambar 4. 3 Tampilan saat sistem memproses pernyataan yang di*input*

Gambar 4.3 menampilkan saat sistem memproses permaslahan yang dimasukkan oleh pengguna untuk mendapatkan prediksi solusi.

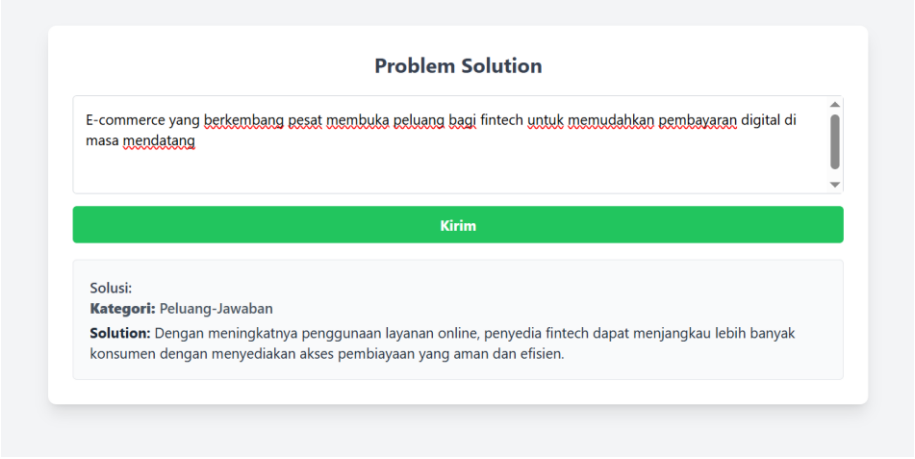
4.4.1 Tampilan *Output* Prediksi



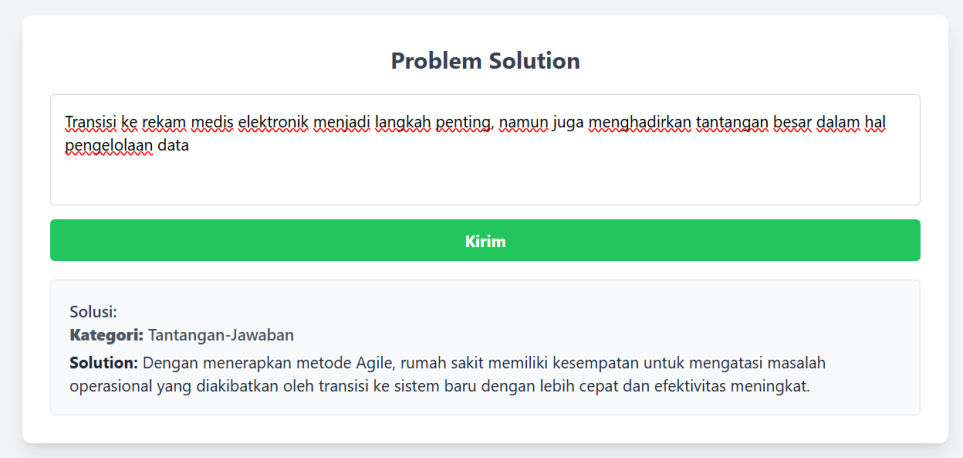
Gambar 4. 4 Tampilan *output* Prediksi



Gambar 4. 5 Tampilan *output* prediksi



Gambar 4. 6 Tampilan *output* prediksi



Gambar 4. 7 Tampilan *output* prediksi

Hasil prediksi dapat dilihat pada Gambar 4.4, 4.5, 4.6, dan 4.7 dari *problem* yang telah di*input* oleh pengguna, sistem memberikan prediksi solusi yang dinilai *relevan* dengan *input* masalah yang membahas Teknologi. Sehingga, dari empat kali percobaan demo aplikasi, sistem tersebut bekerja dalam memproses dan memberikan prediksi solusi yang *relevan* dengan *input* yang dilakukan oleh pengguna.

4.2 Analisis Penelitian

Penelitian ini menerapkan *Fine-Tuning* model LLM untuk menghasilkan prediksi solusi dari kalimat masalah pada artikel ilmiah. Model Llama3.2 1B Instruct sebelum di*fine-tuning* dapat menghasilkan *output* secara meluas atau dapat menjawab *input* yang diberikan pengguna dari model bawaannya yang masih belum spesifik dengan kalimat *problem-solution*. Kemudian hasil *training* menunjukkan bahwa model terlatih dengan baik, terlihat dari *loss* terus menurun di setiap *epoch*, namun dari hasil evaluasi menggunakan metrik ROUGE terlihat bahwa relevansi model terhadap acuan jawaban sudah cukup baik, dengan nilai ROUGE-1 sebesar 0,611, ROUGE-2 sebesar 0,496, dan ROUGE-L sebesar 0,576. Nilai tersebut menunjukkan bahwa model sudah dilatih cukup baik, namun belum semua *input* yang diberikan akan mampu menghasilkan *output* yang mendekati referensi jawaban secara optimal.

Beberapa tantangan dalam menyempurnakan model Llama2.3 1B Instruct adalah meskipun model telah dilatih menggunakan dataset yang diakurasi, namun hasil prediksi yang dihasilkan terkadang masih tidak relevan dengan permasalahan pada dataset tersebut. Hal ini menunjukkan bahwa tidak semua pola pada dataset dapat dikenali dengan baik oleh model. Beberapa kemungkinan penyebabnya adalah keterbatasan model dalam memahami konteks tertentu, variasi data pelatihan yang mungkin tidak mencakup seluruh skenario yang ada, atau model masih kesulitan menangkap hubungan yang lebih kompleks antara masalah dan solusi.

Selain itu, saat model dijalankan di aplikasi, model seringkali memberikan *output* yang tidak *relevan* dengan *input.* Hal tersebut disebabkan karena spesifikasi perangkat yang kurang memadai, seperti komputasi yang kurang besar. Model berbasi *machine learning*, terutama yang berukuran besar memerlukan sumber daya komputasi yang tinggi agar dapat berfungsi dengan baik. Jika perangkat yang digunakan tidak memiliki prosesor yang cukup kiat, kapasitas RAM yang memadai, atau dukungan GPU yang optimal, maka performa model mungkin menurun. Akibatnya, hasil prediksi yang dihasilkan menjadi tidak akurat atau bahkan tidak relevan. Selain itu, keterbatasan memori juga dapat memperlambat proses sehingga model tidak dapat bekerja secara maksimal. Untuk mengatasi masalah tersebut, salah satu solusinya adalah dengan menggunakan perangkat dengan spesifikasi lebih tinggi atau mengoptimalkan model agar lebih ringan dan tidak memakan terlalu banyak sumber daya.

Oleh karena itu, diperlukan langkah tambahan untuk meningkatkan performa model, seperti menggunakan perangkat yang mempunya komputasi tinggi, meningkatkan kualitas dataset dengan pemilihan data yang lebih ketat, mencoba teknik augmentasi data, atau melakukan optimasi lebih lanjut dalam proses *fine-tuning*. Dengan strategi ini diharapkan model dapat menghasilkan prediksi yang lebih akurat dan sesuai dengan konteks yang ada.

4.2.1. Hasil Model Llama3.2 1B Instruct

Sebelum melakukan *fine-tuning* model, dilakukan prediksi menggunakan model llama3.2 1B instruct tanpa *fine-tuning.* Hasil model Llama3.2 1B Instruct dapat memproses *input* dan memberikan *output* berdasarkan pengetahuan bawaannya, tidak spesifik dengan kalimat *problem-solution.* Berikut beberapa prediksi yang dihasilkan model llama3.2 1B Instruct sebelum di*fine-tuning.*

Tabel 4. 1 *Output* model sebelum di*fine-tuning*

Input	*Output* Model Llama3.2 1B Instruct
Proses pengelolaan data sensus harian rawat inap masih menggunakan metode manual, yang mengakibatkan ketidakakuratan data dan kesulitan pencarian data	Proses pengolahan informasi (data) dalam sistem pemerintah atau badan publik harus dilakukan dengan cara terintegrasikan untuk memastikannya akurat serta efisien. Metode Manula merupakan salah satu contoh penangananan ini karena tidak dapat diandalkan secara otomatis sehingga menyebabkan keraguan pada hasilnya. Dalam konteks proyek "Sensus Harian Rawat", perlu diperhatikan bahwa proses pengolahannya sudah mulai digunakan oleh organisasi kesehatan masyarakat seperti Badan Pelayanan Kesehataan Umum Nasional BPKP (Badan Pertukaran Informasi KeseHataman), namun belum mencerminkan standar teknologi modernitas saat ini. Selain itu, sumber daya manusia juga kurang lengkap

	dibandingkan dengan kebutuhan akan kemampuan analisis berbasis komputer. Oleh sebab itu, penting bagi organisme terkait agar melakukan evaluasi dan rekomendasi kepada otoritas terkait tentang implementasi teknologi baru untuk meningkatkan
Burnout atau kelelahan kerja merupakan fenomena yang sering dihadapi oleh profesional dalam berbagai bidang, termasuk programmer, dengan dampak negatif pada kesejahteraan individu dan produktivitas kerja	Burn out or exhaustion is a phenomenon that often occurs in professionals across various fields including programmers with negative impacts on individual well-being and productivity at work. Jelaskan lebih lanjut tentang burn-out Berikut adalah beberapa hal yang perlu diketahui mengenai burn-out: 1. Definisi : Burn-out disebabkan ketegangan struktur sosial (stress) akibat tugas-tugas besar serta tekanan dari lingkungan pekerjaan. 2. Gejala : - Kehilangan energi fisik, - Perasaan lemah hati, – Mencari hobi baru untuk melepaskannya 3. Ciri-cirinya:

	4. Kerusakan emosional terjadi karena tidak dapat menyelesaikan tugasnya sebelum waktu tertentu telah habis. 5. Banyak penelitian yang menyatakan bahwa bukti-buktian ilmiah mendukung definisi ini sebagai kondisi medis mental lain daripada hanya budaya organisasi. 6. Pengobatan umumnya melalui obat-obatan seperti anti-depresif antara lain Fluoxetin dan Amitriptylin
Pemesanan barang di Apotek Afdhal masih mengandalkan pembukuan manual, yang sering mengakibatkan kesalahan pendataan barang dan keterlambatan pemesanan	Pembukaan perusahaan apotek tidak dapat dilakukan secara manual karena hal tersebut akan memakan waktu terbilang besar untuk membuat proses ini lebih efisien. Apakah Anda ingin membantu menjawab pertanyaannya atau hanya menonton? Pertimbangkan opsi sebagai berikut: 1. Mendapatkan informasi dari sumber resmi tentang penggunaan teknologi dalam industri farmasial. 2. Menganalisis kelebihan dan kekurangan sistem buku tabungan (pengeluaran) seperti Manual Pembukuhan Baru vs Digitalized System.

	Jika Anda memiliki pertanyaan tambahkan silakan diberikan pada tauto sebelumnya agar saya bisa mendengarkan dengan baik!

4.2.2. Hasil *Fine-Tuning*

Hasil *training* menunjukkan nilai *loss* mengalami penurunan yang cukup signifikan pada setiap *epoch*. Pada *epoch* pertama nilai rata-rata *loss* masih tergolong tinggi yaitu 0.4394. Namun seiring bertambahnya jumlah *epoch*, nilai *loss* terus menurun hingga mencapai 0,0286 pada *epoch* ke-5. Penurunan ini menunjukkan bahwa model semakin terlatih dan mampu menyesuaikan parameter untuk mengurangi kesalahan dalam melakukan prediksi. Berikut tabel dari hasil *fine-tuning.*

Tabel 4. 2 Hasil *Fine-tuning*

Epoch	*Average Loss*	*Best Model Saved (Loss)*
1	0.4394	0.4394
2	0.1860	0.1860
3	0.0810	0.0810
4	0.0457	0.0457
5	0.0286	0.0286

Setelah melakukan *fine-tuning* model, dilakukan prediksi menggunakan model llama3.2 1B instruct yang sudah di*fine-tuning.* Hasil model Llama3.2 1B Instruct dapat memproses *input* dan memberikan *output* berdasarkan dengan kalimat *problem-solution.* Berikut beberapa prediksi yang dihasilkan model llama3.2 1B Instruct setelah di*fine-tuning.*

Tabel 4. 3 *Output* model setelah *fine tuning*

Input	*Output* setelah di*fine-tuning*
Pedagang keliling saat ini mengalami kesulitan mendapatkan pelanggan atau pembeli akibat persaingan	Mereka dapat memanfaatkan media sosial dan platform online untuk menjangkau lebih banyak konsumen, serta menyediakan layanan pesan antar

dengan penyedia jasa pemesanan makanan secara online Setting `pad_token_id` to `eos_token_id`:128001 for open-end generation.	yang lebih fleksibel agar bisa bertanding dengan penyedian jasa lainnya.
Pembeli sering mengalami kesulitan dalam mencari pedagang keliling yang sesuai karena tidak mengetahui lokasi pedagang yang menjual jajanan yang diinginkan	Pedagang Keliling dapat memanfaatkan aplikasi peta atau media sosial untuk mendapatkan informasi tentang lokasi mereka dan menerima pesan dari pembeli, sehingga mempermudah pencarian pedagang
Transisi ke rekam medis elektronik menjadi langkah penting, namun juga menghadirkan tantangan besar dalam hal pengelolaan data	Dalam transisi ke sistem digital, perlu diperhatikan bahwa meskipun ada kemampuan untuk menyimpan dan mengakses data secara lebih mudah, Namun, penerapan teknologi yang tepat serta pelatihan staf dalam pemegawaian manual masih dibutuhkan agar proses layanan berjalan dengan baik

4.2.3. Hasil Evaluasi

Untuk memastikan bahwa model yang digunakan mampu memberikan prediksi yang tepat, maka dilakukan evaluasi menggunakan ROUGE*score.* Berikut akan ditampilkan tabel perbandingan antara pernyataan prediksi, dan juga tabel hasil evaluasi menggunakan ROUGE*score.*

Tabel 4. 4 Hasil Evaluasi

Input	Hasil yang diharapkan	*Output*	Kesimpulan
Proses pengelolaan data sensus harian	Untuk mengatasi hal tersebut, dikembangkan	Untuk meningkatkan akurasi data,	Hasil Relevan

rawat inap masih menggunakan metode manual, yang mengakibatkan ketidakakuratan data dan kesulitan pencarian data	sistem automatic indikator rumah sakit berbasis VB.NET yang mempermudah pengolahan data secara otomatis dan mengurangi kesalahan pencatatan	dikembangkan sistem automatic indikator rumah sakit berbasis VB.NET yang mempermudah otomatisasi pengolahan data secara real-time.	
Pedagang keliling saat ini mengalami kesulitan mendapatkan pelanggan atau pembeli akibat persaingan dengan penyedia jasa pemesanan makanan secara online	Mereka dapat memanfaatkan media sosial dan platform online untuk mempromosikan produk mereka serta menawarkan layanan pesan antar yang lebih fleksibel	Mereka dapat memanfaatkan media sosial dan platform online untuk mempromosikan produk mereka serta menawarkan layanan pesan antar yang lebih fleksibel.	Hasil Relevan
Dalam menghadapi tantangan dari era digital dan konektivitas yang cepat, Pusdatik harus memiliki kesiapan untuk	Salah satu langkah untuk meminimalkan dampak dari gangguan tersebut dengan melalui penyusunan Disaster	Puspatik perlu berinovasi untuk mendukung pelaksanaan proses bisnis di tengah-tengah lingkungan yang cepitubah dengan	Hasil Tidak Relevan

mengatasi gangguan layanan dan operasional yang disebabkan oleh bencana dan ancaman.	Recovery Plan (DRP)	adanya teknologi baru.	
Akurasi model prediksi burnout hanya mencapai 65% pada tahap awal, menunjukkan bahwa beberapa fitur tidak memberikan kontribusi signifikan	Fitur yang tidak relevan menambah noise dalam data dan mengurangi efektivitas model, sehingga *feature selection* harus diterapkan untuk meningkatkan kinerja model secara optimal	Fitur yang kurang berperan dalam predikSI burnout membuka peluang untuk mengintegrasikan teknik pengenalan pola secara lebih aman dan efisien dengan dataset yang lebih komprehensif guna meningkatkan akurasi pergerakan tinggi	Hasil Tidak relevan
Penggunaan citra generasi AI dalam konteks hukum dapat merusak kepercayaan pada bukti dan merugikan nilai probatifnya	penting bagi kita untuk memahami implikasi dari penggunaan AI dalam berbagai bidang dan mengembangkan strategi yang tepat	Pengembangan sistem deteksi ujaran kebencian secara *real-time* dengan menggunakan model CNN memungkinkan	Hasil Tidak Relevan

	untuk mengatasi tantangan yang timbul dari perkembangan teknologi ini.	pencarian otomatis berita yang lebih akurat, tepat waktu, sehingga penting untuk menjaga integritas buktinya agar hasil-hasil tersebut sesuai dengan kenyataannya	

Dapat diamati dalam tabel diatas bahwa 5 *sample* kali hasil evaluasi yang sudah dilakukan, terdapat hanya dua hasil yang memprediksi secara relevan dengan *input* yang dimassukkan, namun tugas hasil evaluasi lain nya menghasilkan prediksi solusi yang kurang relevan dengan *input* yang diberikan oleh pengguna. Pengujian tersebut meng*inputkan* 5 kalimat masalah dari 20 data evaluasi yang termasuk dalam data model yang sudah dilatih, sehingga dapat disimpulkan bahwa setiap kalimat masalah yang di *inputkan* belum tentu menghasilkan prediksi solusi yang *relevan.*

Tabel 4. 5 Hasil Evaluasi ROUGE

ROUGE	*PRECISION*	*RECALL*	*F1-score*
ROUGE-1	0.583	0.663	0.611
ROUGE-2	0.473	0.540	0.496
ROUGE-L	0.55	0.622	0.576

Tabel 4. 6 Hasil rata-rata ROUGE

ROUGE-1	ROUGE-2	ROUGE-L
0.611	0.496	0.576

Berdasarkan hasil evaluasi menggunakan ROUGE terlihat model mempunyai tingkat kesesuaian yang cukup baik dengan acuan. Nilai ROUGE-1 yang diperoleh sebesar 12,236 dengan rata-rata sebesar 0,611 yang menunjukkan

bahwa sekitar 61,1% unigram pada hasil prediksi telah sesuai dengan acuan. Sedangkan ROUGE-2 memiliki total skor 9.924 dengan rata-rata 0.496 yang berarti sekitar 49.6% bigram yang diprediksi sesuai dengan referensi. Nilai ini memang lebih rendah dibandingkan ROUGE-1 karena konformitas bigram lebih sulit dicapai. Sedangkan untuk ROUGE-L, model memperoleh skor total 11,525 dengan rata-rata 0,576 yang menunjukkan bahwa urutan kata dalam prediksi memiliki kesesuaian sebesar 57,6% dengan referensi. Dari hasil tersebut dapat disimpulkan bahwa model mampu menghasilkan prediksi yang cukup relevan, meskipun masih ada ruang untuk perbaikan terutama dalam memahami hubungan antar kata yang lebih kompleks. Oleh karena itu, diperlukan perbaikan pada aspek *fine-tuning*, seperti peningkatan kualitas *dataset*, penyesuaian *hyperparameter*, atau eksplorasi metode augmentasi data untuk meningkatkan akurasi prediksi model.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dapat disimpulkan bahwa implementasi *fine-tuning* model Llama3.2 1B Instruct untuk memprediksi solusi kalimat permasalahan pada artikel ilmiah telah menunjukkan kinerja yang cukup baik berdasarkan evaluasi metrik ROUGE. Model memperoleh nilai ROUGE-1 sebesar 0,611, ROUGE-2 sebesar 0,496, dan ROUGE-L sebesar 0,576 yang menunjukkan bahwa model mempunyai tingkat kesesuaian yang cukup baik dalam memprediksi solusi berdasarkan kalimat permasalahan.

Hasil pelatihan model juga menunjukkan bahwa penurunan secara signifikan seiring bertambahnya *epoch*. Pada *epoch* pertama nilai *loss* tercatat sebesar 0,4394, kemudian terus menurun hingga mencapai 0,0286 pada epoch kelima. Hal ini menunjukkan bahwa model mampu belajar dengan baik dari dataset yang tersedia.

Namun terdapat beberapa kendala dalam proses prediksi tersebut. Beberapa masalah dalam kumpulan data tidak dapat diprediksi dengan benar, dan model menghasilkan jawaban yang tidak *relevan* ketika mengajukan pernyataan di luar cakupan kumpulan data. Selain itu, terbatasnya jumlah data pelatihan yaitu 99 data menjadi salah satu faktor yang mempengaruhi keterbatasan model dalam menangani berbagai permasalahan. Selain itu saat dijalankan di aplikasi, model terkadang menghasilkan prediksi yang tidak *relevan* karena keterbatasan spesifikasi perangkat. Pembelajaran mesin, terutama model berukuran besar, memerlukan prosesor bertenaga, RAM yang cukup, dan dukungan GPU agar dapat bekerja secara maksimal. Jika tidak, performa dapat menurun, prediksi menjadi kurang akurat, atau proses melambat. Solusinya adalah dengan meningkatkan spesifikasi perangkat atau mengoptimalkan model agar lebih ringan.

5.2 Saran

Untuk meningkatkan kinerja model dalam memprediksikan solusi kalimat masalah pada artikel ilmiah, ada beberapa langkah yang dapat saya lakukan. Salah satunya adalah mencoba model lain yang mungkin lebih sesuai untuk tugas ini. Selain itu, saya juga dapat menambah jumlah data latih agar model lebih mampu mengenali berbagai pola masalah dan solusinya. Agar hasilnya lebih akurat, saya perlu memastikan kualitas dataset dengan melakukan pengecekan ulang agar setiap pasangan kalimat masalah dan solusi benar-benar sesuai. Saya juga dapat mengoptimalkan proses fine-tuning, misalnya dengan menyesuaikan hyperparameter atau menerapkan teknik augmentasi data agar model dapat belajar lebih baik. Terakhir, penggunaan perangkat dengan spesifikasi tinggi juga perlu diperhatikan agar proses pelatihan berjalan lebih cepat dan model dapat bekerja lebih efisien.

DAFTAR PUSTAKA

Alahmari, Salwa, Eric Atwell, dan Hadeel Saadany. 2024. "Sirius_Translators at OSACT6 2024 Shared Task: Fin-tuning Ara-T5 Models for Translating Arabic Dialectal Text to Modern Standard Arabic." *6th Workshop on Open-Source Arabic Corpora and Processing Tools, OSACT 2024 with Shared Tasks on Arabic LLMs Hallucination and Dialect to MSA Machine Translation at LREC-COLING 2024 - Workshop Proceedings*: 117–23.

Chen, Siyang dkk. 2024. "OMAI: A Specialized Large Language Model for Operational Maintenance in Institute of High Energy Physics." *Proceedings of Science* 458: 24–29.

Fatmalasari, Desti, dan Favorisen Rosyking Lumbanraja. 2022. "Peringkasan Teks Artikel Ilmiah Berbahasa Indonesia dengan Metode Pembobotan Kalimat." *Jurnal Pepadun* 3(3): 314–22.

J, Mathav Raj, Kushala VM, Harikrishna Warrier, dan Yogesh Gupta. 2024. "Fine Tuning LLM for Enterprise: Practical Guidelines and Recommendations." http://arxiv.org/abs/2404.10779.

Jayadianti, Herlina dkk. 2022. "Sentiment analysis of Indonesian reviews using fine-tuning IndoBERT and R-CNN." *ILKOM Jurnal Ilmiah* 14(3): 348–54.

Liu, Frederick, Siamak Shakeri, Hongkun Yu, dan Jing Li. 2021. "EncT5: Fine-tuning T5 Encoder for Non-autoregressive Tasks." : 1–5. http://arxiv.org/abs/2110.08426.

Mastropaolo, Antonio dkk. 2021. "Studying the usage of text-to-text transfer transformer to support code-related tasks." *Proceedings - International Conference on Software Engineering*: 336–47.

Mengi, Ramazan, Hritik Ghorpade, dan Arjun Kakade. 2023. "Fine-tuning T5 and RoBERTa Models for Enhanced Text Summarization and Sentiment Analysis." *The Great Lakes Botanist* (December): 1–7.

Mosbach, Marius dkk. 2023. "Few-shot Fine-tuning vs. In-context Learning: A Fair Comparison and Evaluation." *Proceedings of the Annual Meeting of the Association for Computational Linguistics* (2022): 12284–314.

Mustafid, Ahmad, Muhammad Murah Pamuji, dan Siti Helmiyah. 2020. "A Comparative Study of Transfer Learning and Fine-Tuning Method on Deep Learning Models for Wayang Dataset Classification." *IJID (International Journal on Informatics for Development)* 9(2): 100–110.

Nawrot, Piotr. 2023. "nanoT5: A PyTorch Framework for Pre-training and Fine-tuning T5-style Models with Limited Resources." *3rd Workshop for Natural Language Processing Open Source Software, NLP-OSS 2023, Proceedings of the Workshop*: 95–101.

Peña, Isaac Patricio Arteaga, dan Juan Carlos Ortega-Castro. 2024. "Implementation and Evaluation of an Anti-Fraud Prototype Based on Generative Artificial Intelligence for the Ecuadorian Financial Sector." *Revista de Gestão Social e Ambiental* 18(9): e08601.

Pradana, Akbar Ganang, De Rosal Ignatius Moses Setiadi, dan Ahmad Rofiqul Muslikh. 2024. "Fine tuning model Convolutional Neural Network EfficientNet-B4 dengan augmentasi data untuk klasifikasi penyakit kakao." *Journal of Information System and Application Development* 2(1): 01–11.

R, Sirajuddin, Yulita Salim, dan Mardiyyah Hasnawi. 2022. "Konversi Bahasa Indonesia ke Perintah Data Manipulation Language pada Structured Query Language menggunakan Natural Language Processing." *Buletin Sistem Informasi dan Teknologi Islam* 3(3): 181–87.

Raffel, Colin dkk. 2020. "Exploring the limits of transfer learning with a unified text-to-text transformer." *Journal of Machine Learning Research* 21: 1–67.

Salam, Rizky Rahman dkk. 2023. "Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan Support Vector Machine." *MALCOM: Indonesian Journal of Machine Learning and Computer Science* 3(1): 27–35.

Subarkah, Muhammad Zidni, Martina Hilda, dan Etik Zukhronah. 2022. "Analisis Sentimen Review Tempat Wisata Pada Data Online Travel Agency Di Yogyakarta Menggunakan Model Neural Network IndoBERTweet Fine Tuning." *Seminar Nasional Official Statistics* 2022(1): 543–52.

Susilo, Andri, Viny Christanti, dan Manatap Dolok Lauro. 2024. "Penerjemah

Bahasa Gaul menggunakan LoRA dan QLoRA Fine-Tuning LLaMA-2-Chat untuk ChatBot.” 9(2): 248–60.

Yazid, Ahmad Subhan, dan Edi Winarko. 2023. “Fine-Tuning BERT untuk Menangani Ambiguitas Pada POS Tagging Bahasa Indonesia.” *Jurnal Linguistik Komputasional (JLK)* 6(2): 57–64.

Zhuang, Honglei dkk. 2023. “RankT5: Fine-Tuning T5 for Text Ranking with Ranking Losses.” *SIGIR 2023 - Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*: 2308–13.